

CLAIMS

What is claimed is:

1. A method of prioritizing content, said method comprising:
receiving at a proxy cache a request for a first content object;
searching a data structure of said proxy cache for a portion of said first content object;
calculating the size of said portion of said first content object on said proxy cache;
deriving the normally utilized size of said first content object requested; and
comparing the size of said portion of said first content object on said proxy cache with the normally utilized size of said first content object requested.
2. The method of Claim 1 comprising:
receiving said request for said first content object from a client device.
3. The method of Claim 1 wherein the calculating the size of said portion of said first content object on said proxy cache comprises:
counting a total number of segments of said portion of said first content object cached on said proxy cache;
measuring a length of said total number of segments of said portion of said first content object; and
multiplying said total number of segments with said length of said segments of said first content object.
4. The method of Claim 1 wherein the deriving the normally utilized size of said first content object requested comprises:
utilizing said access log of said proxy cache to establish an average view length of said first content object.
5. The method of Claim 1 wherein said method comprises:
admitting no further portion of said first content object to said proxy cache if said normally utilized size is less than the size of said portion of said first content object on said proxy cache.
6. The method of Claim 1 wherein said method comprises:

admitting a further segment of said first content object to said proxy cache if said normally utilized size is greater than the size of said portion of said first content object on said proxy cache.

7. The method of Claim 6 comprising:
checking available cache space on said proxy cache for room to admit said further segment of said first content object; and
making room on said proxy cache for said further segment of said first content object.

8. The method of Claim 7 wherein said making room on said proxy cache comprises:
calculating a utility value for other content objects on said proxy cache;
prioritizing said other content objects based on said utility value;
selecting one of said other content objects with a smallest utility value;
removing a lowest priority portion of said other content object with the smallest utility value on said proxy cache; and
admitting said further segment of said first content object to said proxy cache.

9. The method of Claim 8 wherein if said one of said other content objects with the smallest utility value is fully cached further comprises:
segmenting said one of said other content objects with the smallest utility value based on a previous duration of access to said one of said other content objects with the smallest utility value.

10. The method of Claim 8 wherein if said one of said other content objects with the smallest utility value is not fully cached further comprises:
removing all segments of said one of said other content objects with the smallest utility value.

11. A caching proxy comprising:
a communication link to a content source and a communication link to a client device;
a memory unit coupled to said communication links; and
a processor coupled to said memory unit, said processor for executing a method of prioritizing content, said method comprising:
receiving at a proxy cache a request for a first content object;

searching an access log of said proxy cache for said first content object;

checking available cache space on said proxy cache for space to admit said first content object; and

admitting all of said first content object to said proxy cache.

12. The caching proxy of Claim 11 wherein said method comprises: calculating a utility value for other content objects on said proxy cache; prioritizing said other content objects based on said utility value; selecting a second content object with a smallest utility value; and removing a lowest priority portion of said second content object from said proxy cache.

13. The caching proxy of Claim 12 wherein if said second content object is not segmented, said method comprising:

utilizing said access log of said proxy cache to establish an average view length of said second content object; and

dividing said second content object into segments collaborate with said average view length.

14. The caching proxy of Claim 12 comprising:

removing all segments of said second content object with the smallest utility value if said second content object with the smallest utility value is not fully cached.

15. The caching proxy of Claim 11 wherein said method comprises:

receiving said request for said first content object from an end user.

16. A computer-usable medium having computer-readable program code embodied therein for causing a caching proxy to perform a method of prioritizing content, said method comprising:

receiving at a proxy cache a request for a first content object;

searching an access log of said proxy cache for said first content object;

finding at least one segment of said first content object on said proxy cache;

calculating the size of said first content object on said proxy cache;

calculating the average view length of the segment of said first content object requested;

comparing the size of said first content object on said proxy cache and the average view length of said first content object requested;

denying a further segment of said first content object to said proxy cache if said average view length is less than the size of said first content object on said proxy cache; and

admitting a further segment of said first content object to said proxy cache if said average view length is greater than the size of said first content object on said proxy cache.

17. The computer-usable medium of Claim 16 wherein said computer-readable program code embodied therein causes a caching proxy to perform a method of prioritizing content, said method comprising:

receiving said request for said first content object from an end user.

18. The computer-usable medium of Claim 16 wherein said computer-readable program code embodied therein causes a caching proxy to perform a method of prioritizing content, said method comprising:

finding a total number of said at least one segment of said first content object cached on said proxy cache;

measuring a length of said at least one segment of said first content object; and

multiplying said total number of cached segments with said length of said at least one segment of said first content object.

19. The computer-usable medium of Claim 16 wherein said computer-readable program code embodied therein causes a caching proxy to perform a method of prioritizing content, said method comprising:

utilizing said access log of said proxy cache to establish an average view length of said first content object.

20. The computer-usable medium of Claim 16 wherein said computer-readable program code embodied therein causes a caching proxy to perform a method of prioritizing content, said method comprising:

checking available cache space on said proxy cache for room to admit said further segment of said first content object;

calculating the utility value for other content objects on said proxy cache;

prioritizing said other content objects based on said utility value;

selecting one of said other content objects with the smallest utility value;

removing portions of lowest priority of said other content objects with the smallest utility value on said proxy cache; and

admitting said further segment of said first content object to said proxy cache.

21. The computer-usable medium of Claim 20 wherein said computer-readable program code embodied therein causes a caching proxy to perform a method of prioritizing content, said method comprising:

segmenting said one of said other content objects with the smallest utility value based on a previous duration of access to said one of said other content objects with the smallest utility value if said one of said other content objects with the smallest utility value is fully cached; and

removing all segments of said one of said other content objects with the smallest utility value if said one of said other content objects with the smallest utility value is not fully cached.